3D Object Proposals using Stereo Imagery for Accurate Object Class Detection

Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Huimin Ma, Sanja Fidler and Raquel Urtasun

Presentation by Jungwook Lee

Why use proposals?

- Smart proposal generation methods helps in reduce the search space

- High recall contributes to higher accuracy for overall detection

- Current deep neural networks have very high performance on classification

- 3D vs. 2D Proposals (occlusion, scale variation)

3D Object Proposal Generation

- Proposal Generation as Energy Minimization

$$E(\mathbf{x}, \mathbf{y}) = \mathbf{w}_{c,pcd}^{\top} \phi_{pcd}(\mathbf{x}, \mathbf{y}) + \mathbf{w}_{c,fs}^{\top} \phi_{fs}(\mathbf{x}, \mathbf{y})$$
$$+ \mathbf{w}_{c,ht}^{\top} \phi_{ht}(\mathbf{x}, \mathbf{y}) + \mathbf{w}_{c,ht-contr}^{\top} \phi_{ht-contr}(\mathbf{x}, \mathbf{y}).$$

Point Cloud Density

- Measure of how dense is a bounding box with point clouds

$$\phi_{pcd}(\mathbf{x}, \mathbf{y}) = \frac{\sum_{v \in \Omega(\mathbf{y})} P(v)}{|\Omega(\mathbf{y})|}$$

Free Space

- Potential term to encourage less free space within the box

$$\phi_{fs}(\mathbf{x}, \mathbf{y}) = \frac{\sum_{v \in \Omega(\mathbf{y})} (1 - F(v))}{|\Omega(\mathbf{y})|}$$

Height Prior

- Potential which uses known average class height

$$\phi_{ht}(\mathbf{x}, \mathbf{y}) = \frac{1}{|\Omega(\mathbf{y})|} \sum_{v \in \Omega(\mathbf{y})} H_c(v)$$

with

$$H_c(v) = \begin{cases} \exp\left[-\frac{1}{2}\left(\frac{d_v - \mu_{c,ht}}{\sigma_{c,ht}}\right)^2\right], & \text{if } P(v) = 1 \\ 0, & \text{o.w.} \end{cases}$$

Height Contrast

- Potential that uses the fact surrounding box should have lower values of height relative to the “class box”

$$\phi_{ht-contr}(\mathbf{x}, \mathbf{y}) = \frac{\phi_{ht}(\mathbf{x}, \mathbf{y})}{\phi_{ht}(\mathbf{x}, \mathbf{y}^{+}) - \phi_{ht}(\mathbf{x}, \mathbf{y})}$$

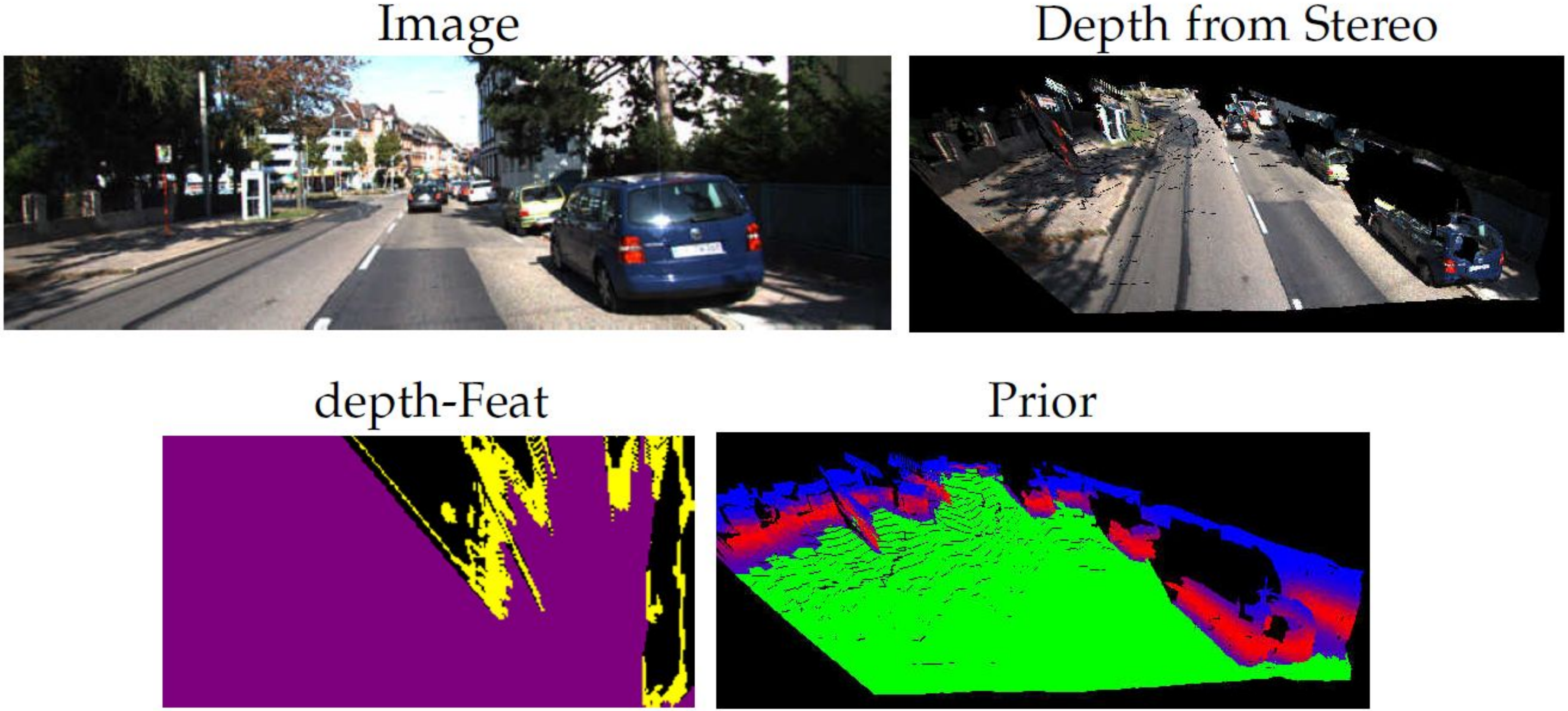
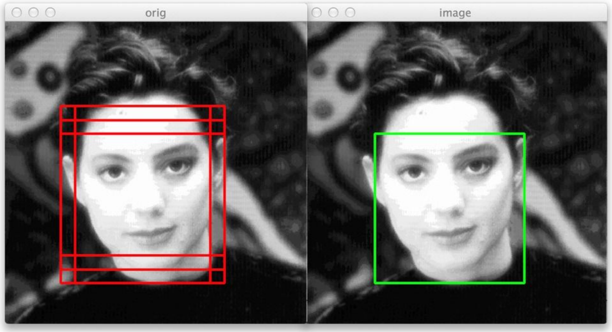


Image
Depth from Stereo
depth-Feat
Prior

Inferencing

Steps:

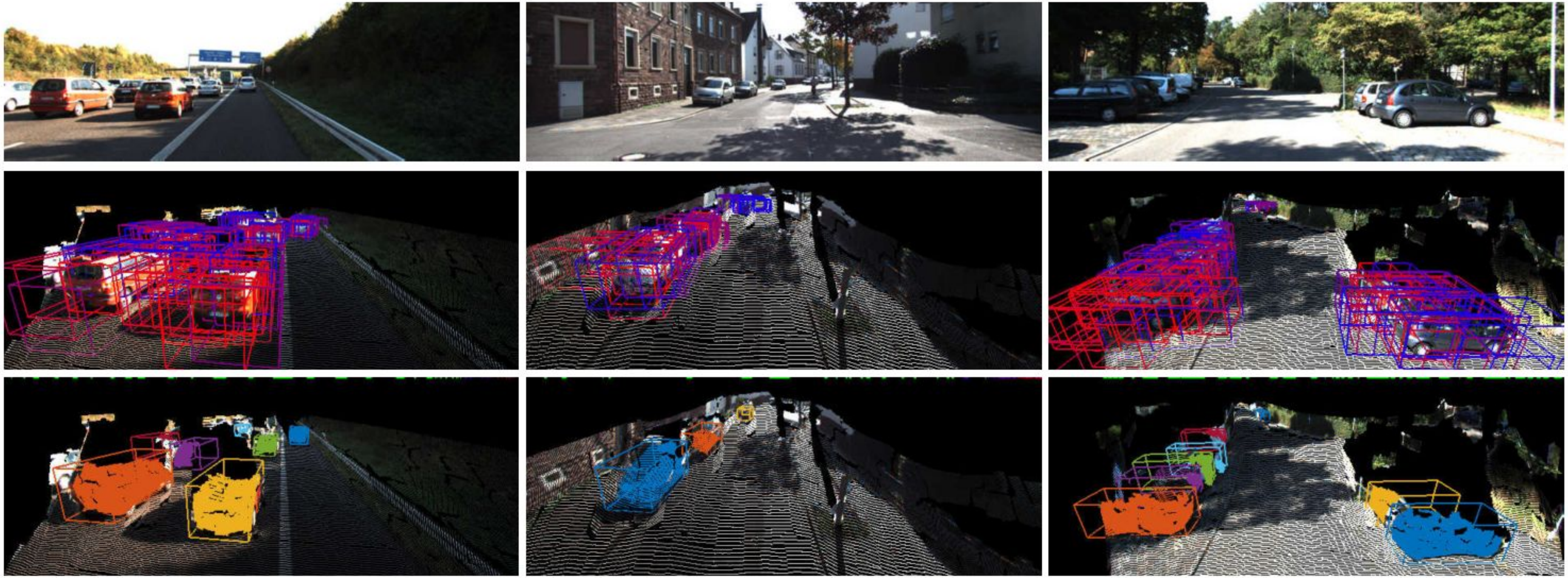
1) Compute $\mathbf{x}$, Discretize 3D space, Ground plane estimation
2) Candidate box sampling (along ground plane, skip empty boxes)
3) Exhaustive scoring based on $E(\mathbf{x}, \mathbf{y})$
4) NMS to obtain top K **diverse** 3D proposals



Greedy Selection Algorithm

$$\mathbf{y}^m = \underset{\mathbf{y} \in \mathcal{Y}}{\operatorname{argmin}}\, E(\mathbf{x}, \mathbf{y})$$

$$\text{s.t.} \quad \mathrm{IoU}(\mathbf{y}, \mathbf{y}^i) < \delta, \quad \forall i \in \{0, \ldots, m-1\}$$



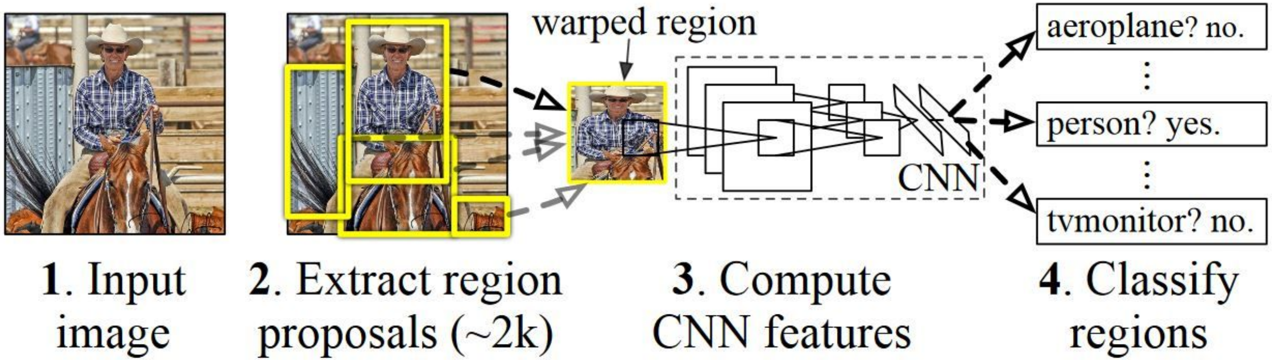
3D Object Detection

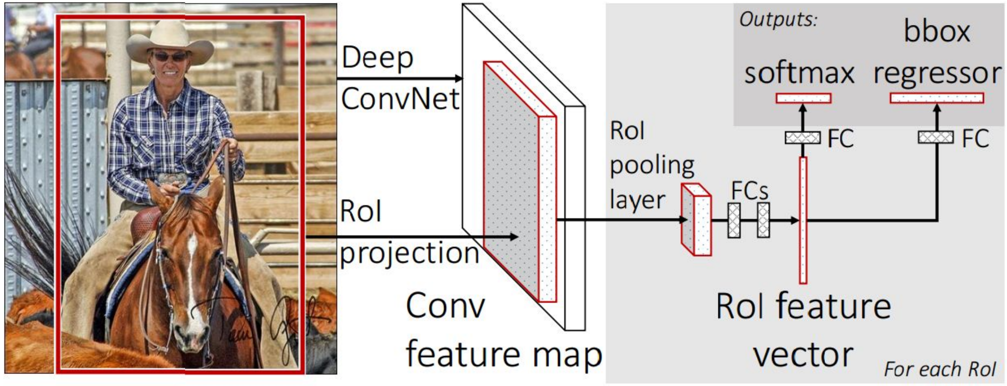
Input : top-ranked 3D object proposals, stereo image (RGB, HHA)

Output: Bounding Box Regression Parameters, Class Score, Orientation

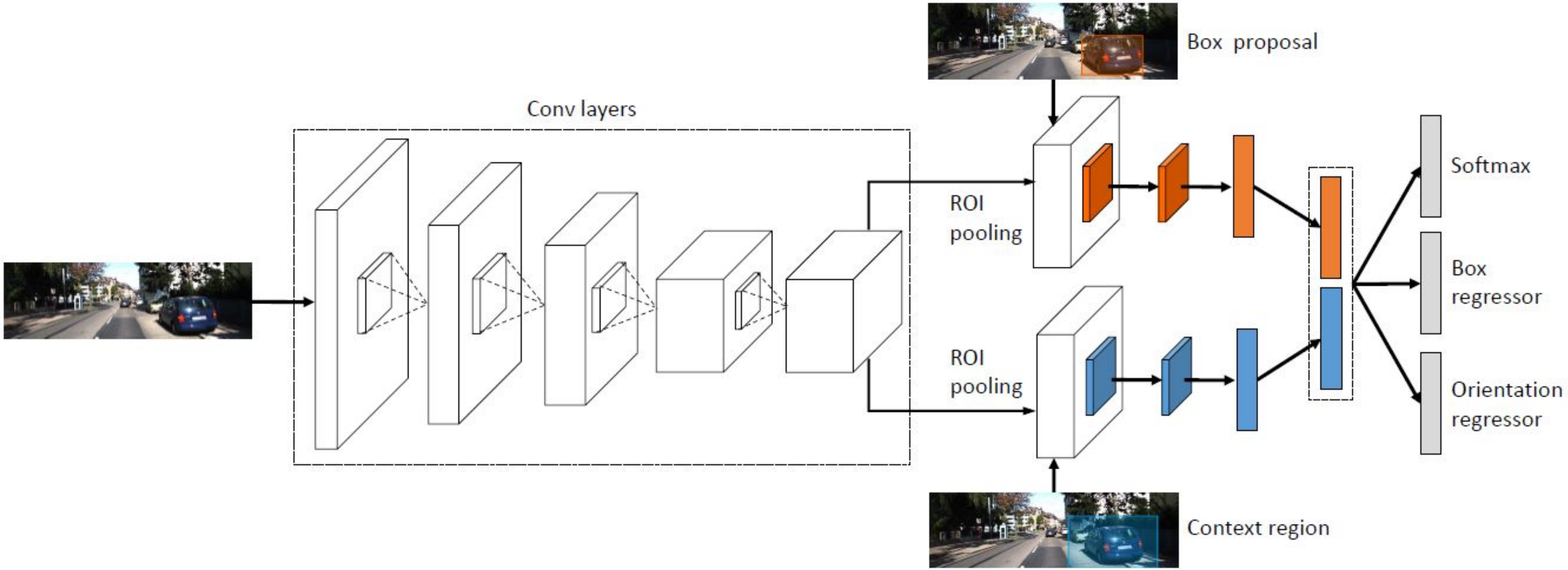
- Deep Neural Networks: Convolutional Networks (cs231n)

- Based on R-CNN variant, Fast R-CNN

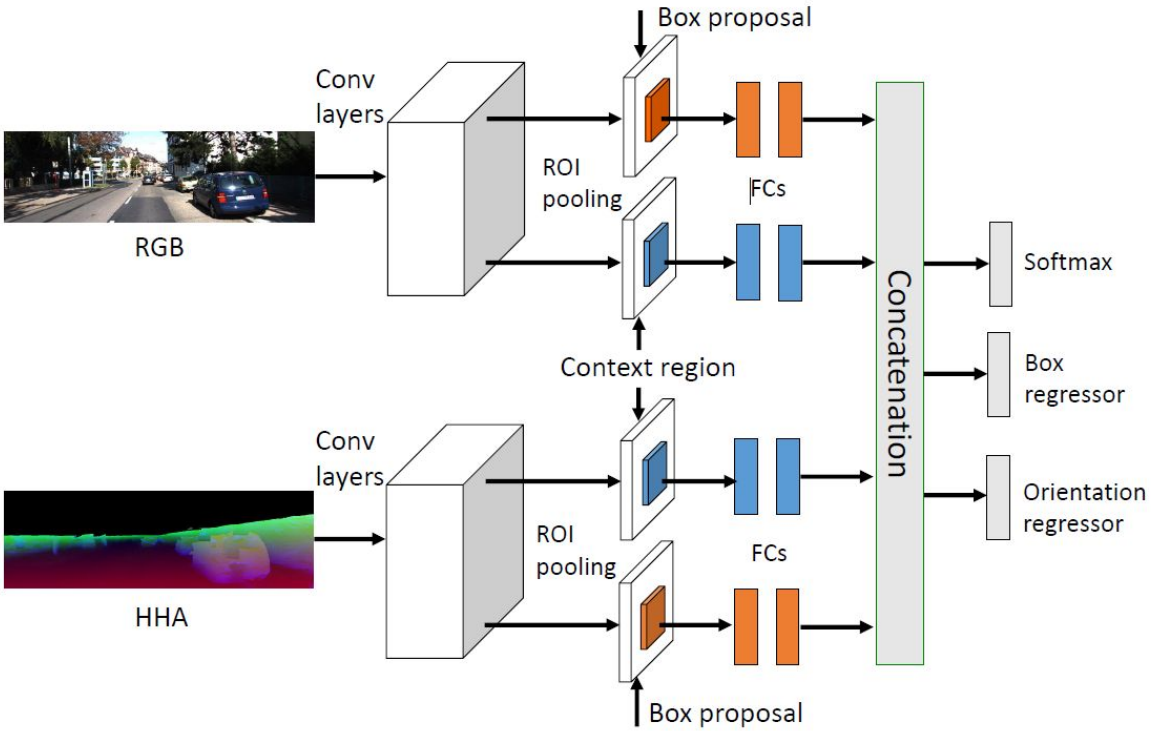

warped region
aeroplane? no.
person? yes.
tvmonitor? no.
CNN
1. Input image
2. Extract region proposals (~2k)
3. Compute CNN features
4. Classify regions


Deep ConvNet
RoI projection
Conv feature map
RoI pooling layer
FCs
Outputs:
softmax
bbox regressor
FC
FC
RoI feature vector
For each RoI

2D Detection Architecture



3D Detection Architecture



Performance Measures

- Proposal Recall: Measure of how much of the objects that the proposals extract from the ground truth set.
- Precision: Measure of how many of the actual positive detection are indeed true objects.

$$R_{OB} = \frac{\text{N.o. correctly detected rectangles}}{\text{N.o. rectangles in the database}}$$

$$P_{OB} = \frac{\text{N.o. correctly detected rectangles}}{\text{Total n.o. detected rectangles}}$$

Performance Measures

- Average Precision (2D, 3D), Average Localization Precision

$$AP = \frac{1}{11} \sum_{r \in \{0, 0.1, \ldots, 1\}} p_{interp}(r)$$

$$p_{interp}(r) = \max_{\tilde{r}:\tilde{r} \geq r} p(\tilde{r})$$

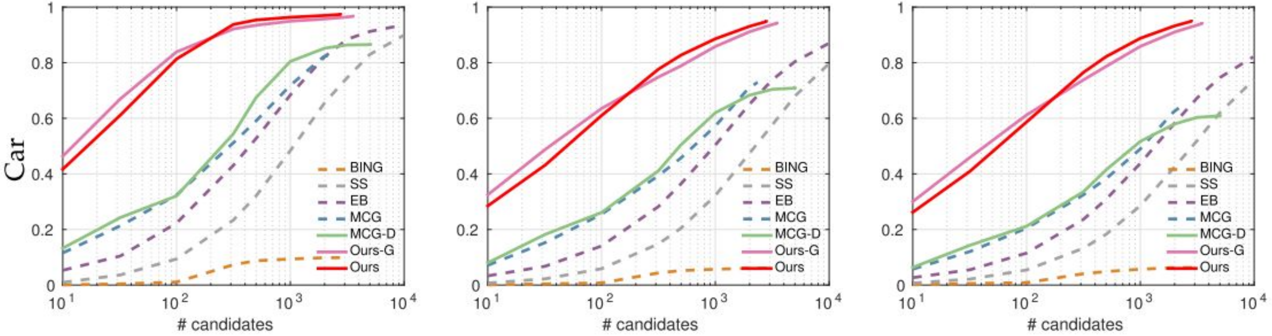
Performance Measures

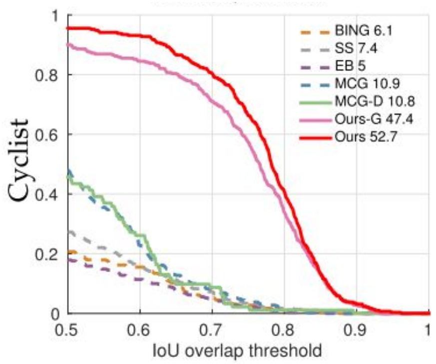
- Average Orientation Similarity

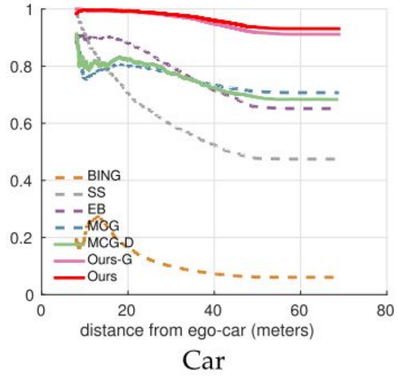
$$AOS = \frac{1}{11} \sum_{r \in \{0,0.1,..,1\}} \max_{\tilde{r}:\tilde{r} \geq r} s(\tilde{r})$$

$$s(r) = \frac{1}{|\mathcal{D}(r)|} \sum_{i \in \mathcal{D}(r)} \frac{1 + \cos \Delta_\theta^{(i)}}{2} \delta_i$$

Proposal Recall Results (2D)

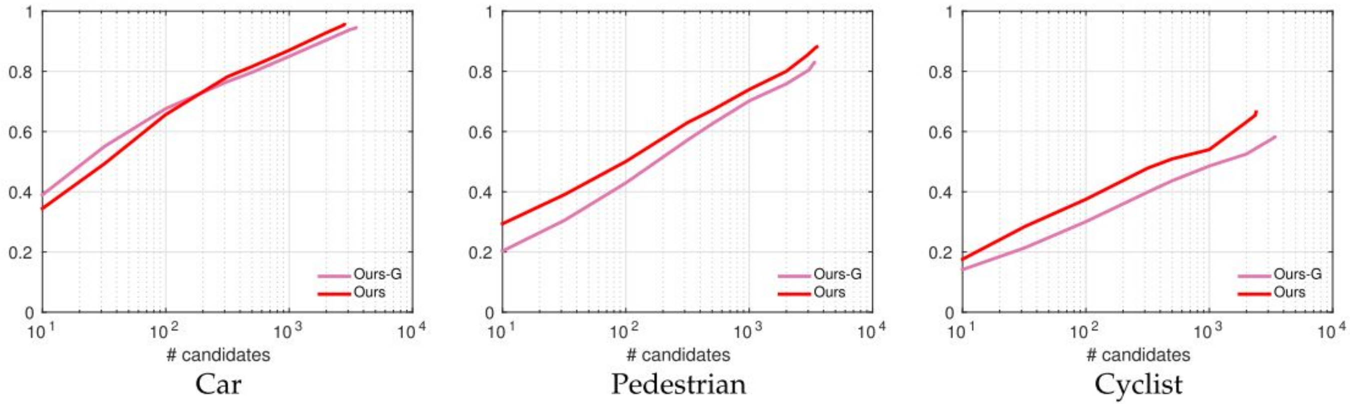






Proposal Recall Results (3D)

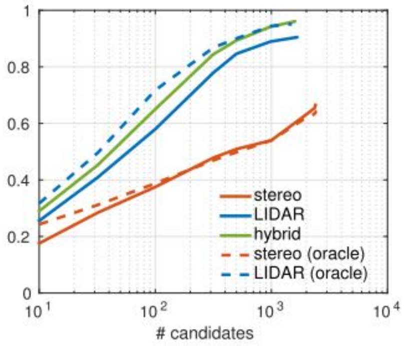
- 0.25 IoU, moderate data



- Proposal Generation Runtime: ~ 2s for 2K proposals

Summary of Key Results

- Hybrid approach using Lidar:
  - stereo PC for road region classification
  - lidar point for plane fitting and inferencing
- Proposal Recall:
  - Hybrid good for small objects (pedestrian, cyclist) and far objects.
  - Highest 3D Recall with Hybrid, but 2D Recall is better with stereo.
- Detection and Localization:
  - Stereo works best on 2D detection and Easy set for 3D detection.
  - Hybrid is best combination for 3D tasks on Moderate and Hard sets (Highest AP, ALP).



Cyclist

- Network design
    - Joint BB and OR (multi-task loss) results in boost in AOS, not much for AP(2D)
- Contextual branch
    - Highest 2D AP and AOS for car. (by small margin)
    - Claims for pedestrian and cyclist, didn’t work out due to the number of weights (2x model for contextual branch and limited data for pedestrian and cyclist)
- RGB-HHA stream
    - RGB-HHA requires more GPU memory, so used 7-layer VGG ConvNet weights
    - Improvement for both 2D (~0.5%) and 3D detection (~ 5-10%) than just RGB
    - 3D detection highest at 7 layer RGB-HHA with hybrid, (better than 16 layer RGB input)
- Ground Plane
    - Using ground truth planes didn’t improve much for stereo
    - Only improves pure lidar approaches. (Good ground plane estimation needed for pure lidar based detection)

TABLE 4: **Object detection (top)** and **orientation estimation (bottom) results on KITTI's validation set**. Here, ort: orientation regression loss; ctx: contextual information; cls: class-specific weights in proposal generation. All methods use 2K proposals per image. VGG-16 network is used.

Metric	Method	ort	ctx	cls	Cars			Pedestrians			Cyclists		
					Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
AP_{2D}	SS [7]		-		75.91	60.00	50.98	54.06	47.55	40.56	56.26	39.16	38.83
	EB [11]				86.81	70.47	61.16	57.79	49.99	42.19	55.01	37.87	35.80
	Ours			✓	92.18	87.26	78.58	72.56	69.08	61.34	**90.69**	62.82	58.26
		✓		✓	92.67	87.52	78.78	72.42	**69.42**	61.55	85.92	62.54	57.71
		✓	✓		92.76	87.30	78.61	**73.76**	66.26	**63.15**	85.91	62.82	57.05
		✓	✓	✓	**93.08**	**88.07**	**79.39**	71.40	64.46	60.39	83.82	**63.47**	**60.93**
AOS	SS [7]		-		73.91	58.06	49.14	44.55	39.05	33.15	39.82	28.20	28.40
	EB [11]				83.91	67.89	58.34	46.80	40.22	33.81	43.97	30.36	28.50
	Ours			✓	39.52	38.24	34.01	34.15	33.08	29.27	63.88	43.85	40.36
		✓		✓	91.46	**85.80**	76.73	**62.25**	**59.15**	**52.24**	**77.60**	**55.75**	51.23
		✓	✓		91.22	85.12	75.74	61.62	55.01	52.14	74.28	53.96	49.05
		✓	✓	✓	**91.58**	**85.80**	**76.80**	61.57	54.79	51.12	73.94	55.59	**53.00**

Contributions

- Spatial information is far more important than appearance for generating good proposals and detection/localization in 3D
  - Deep hierarchical appearance features <<<< spatial features for 3D proposals
  - HHA, which encodes spatial information, significantly improves overall 3D detection

- Proposal Generation for hard objects
  - Even if sparse, very useful in terms of proposal generation for **Small** and **Far** objects (lidar accuracy > density of data)

Shortcomings/Improvements

- Handcrafted features -> Can DNN learn these features? (RPN)
- Knowledge of the prior data
- Relies a lot on pre-processed data (Stereo Disparity, Ground plane)
- Not yet fast enough for on-road detection.
  (~0.83 hz for proposals only, 0.5 hz for forward pass)
- Increase in model size (context) to performance is questionable
- Kitti has no 3D detection test -> contribution for our own dataset.
- Lots of room for improvement in 3D detection for cyclists